EASYCOM

The client/server technology for AS/400

Advanced Client Easycom/400 for Delphi

AURA Equipements

MIS Systems

User's Manual

Ref: DP-USER$0398

User License

OWNERSHIP OF THE SOFTWARE

1. The Advanced Client Easycom/400 for Delphi software programs (« Software ») and accompanying written materials (the « Documentation ») are owned by MIS Systems and AURA Equipements (« Licensors ») and are protected by International copyright laws and by international treaties. The Easycom engine is a product that is the sole propriety of AURA Equipements.

DEFINITIONS

2. End User: The End user is either the natural person or the legal subject that bought the license or that uses an Evaluation Version.

3. Evaluation Version: Your use of the software is for the purpose of evaluating whether to purchase an ongoing license to the Software. You use an Evaluation Version when you install the Software without the appropriate serialization code (which takes the form of a serial number) to be install system. When you use an Evaluation Version, as soon as the DLL files in the Software are called upon, will be prompted before a dialog box appears explaining that you are running an Evaluation Version of the Software. You will then receive about twenty minutes of time to use the product by a similar dialog box every minute after one hour of use. The total number of time you can use the DLL is limited to 100. Anyhow the case the evaluation period for use is limited to 15 days.

4. Software: Software is the right to use the software-product "EASYCOM Server" and the Advanced Client Easycom/400 for Delphi a direct level access for a PC Side for one development tool.

(a) Installation: The " Advanced Client Easycom/400 for Delphi" software, includes two parts: the client part and the server part. The client part is to be installed on the PC which one is linked to the AS/400. The server part is to be installed on the AS/400.
(b) You can install the server part only once at a time one a specific AS/400 machine. (The software activation key is calculated by using the AS/400 serial number). In case, the software is to access to several AS/400, you have to purchase a license for each AS/400.
(c) The client part can be installed on as many PC's you want. But only as many PC's as the number of users of the license will be able to access the AS/400 at a time. (The activation key is calculated by using the number of users too)
(d) The evaluation version includes only one server part, one client part with one single connection.
4.1 The " Advanced Client Easycom/400 for Delphi developer" gives to the developer all the functions requested for the development of an application.
4.2 The " Advanced Client Easycom/400 for Delphi runtime" gives to the user the possibility to execute the application, which has been developed with " Advanced Client Easycom/400 for Delphi developer". You can not develop any application with " Advanced Client Easycom/400 for Delphi runtime".
4.3 The evaluation version is " Advanced Client Easycom/400 for Delphi developer" limited for a period of time.

5. Maintenance release: shall mean corrections of errors and minor additions or improvements of functions in the software that are released to the public by Licensor from time to time at its sole discretion. A new maintenance release is identified by a modification of the product numbering to the right of a decimal point (e.g., a released labeled « 4.1 » would be a maintenance release of version 4 or 4.0 of the software).

6. New Version: shall mean important additions of improvements of functions or modules in the software that are released to the public by Licensor from time to time at its sole discretion. A New Version is identified by a modification of the product numbering to the left of decimal point (e.g., a move from « 4.1 » to « 5.0 »). Licensor maintains full discretion as to whether and when to issue a new release of the Software and whether to classify a new release as a « Maintenance Release » or « New Version ».

GRANT OF LICENSE

7. License for Evaluation: You may use the Evaluation Version of the Software and Documentation in accordance with the terms of this Agreement (except that you will not benefit from the rights set forth in sections 9,18,22 and 23 below) and solely for the purpose of deciding whether to purchase a full License to the Software. You may also reproduce and distribute, including by means of posting on an Internet forum, the un-installed Evaluation Version of the Software known as the « install disks », provided you do not modify or delete this agreement, any copyright or trademark notices or any portion of the Software. You shall be solely responsible for any costs or liabilities arising from any such reproduction or distribution.

8. Acquisition of Full License: If you have an Evaluation Version of the Software and wish to benefit from the rights set forth in sections 9,18,22 and 23 below, you must obtain a serialization code, which will disable the evaluation dialog prompt cycle described in section 3 above. A Serialization code may be obtained, in exchange for the applicable Licensee fee, simply by contacting Licensors or any authorized AURA Equipements distributors, the coordinates of which are set forth below.

9. License: In consideration for the payment of the License fee and your agreement to abide by the terms of this agreement, Licensor grants non exclusive License to use the Software and the documentation depending on (i) number of Client Licenses bought (ii) the price paid Software Development and/or Software Run Time, connected to a single server.

Governing Law: This agreement shall be governed by the laws of the United States and all the signature of the international copyright treaties.

Reference: DP-USER$0398

TABLE OF CONTENTS

Introduction 9

Performance 10

- Advanced Client Easycom/400 for Delphi database-related features include 10
- ACE/400 for Delphi resource-related features include 10
- AS/400 Communication Protocol 11
  - APPC 11
  - Microsoft SNA/Server 11
  - TCP/IP 11
- Native Mode 11
- Batch Reads 11

Integration 11

How does it work? 12

- Server Side 12
- Client Side 12

AS/400 Extensions 14

- EASYCOM Integration and Interoperability 14
- Record-Oriented 14
- Co-operative Systems 14
- Incremental Phase-in 14

System Requirements 19

- AS/400 System Requirements 19
- PC System Requirements 19
- Product features 19

AS/400 Installation Steps for APPC 23

- Important notices: 23

AS/400 Installation Steps for TCP/IP 23

AS/400 Activating the Serialization key 27

- AS/400 Name: 27
- Server Program: 28
- Network Type: 28
- NoWait Mode: 28

Local LU Alias: 29
Delphi Installation Steps 29
Alias configuration options 30
SERVER NAME 30
LIBRARY NAME 30
BROWSE LIBS 30
LANGDRIVER 30
CODEPAGE FILE 30
CMTCTL 31
PRIVATE SESSION 31
LOG LEVEL 31
LOG FILE 31
Misc. configuration options 31
Important note: 32
Testing Data Access 36
Field type management 38
Running the Sample Programs 41
Delphi\MasDetTb\ (16 and 32 bit): 41
Delphi\MasDetTQ\ (16- and 32-bit): 43
Delphi\LookUp\ (16- and 32-bit): 44
Delphi\FlyIndex\ (16- and 32-bit): 45
Delphi32\Filters (32-bit): 46
Delphi\Find (16 and 32 bit): 47
Delphi\StProc (16 and 32 bit): 48
Delphi3\DCube (32 bit, Delphi 3 only): 49
Delphi3\NoLogIn 51
Delphi3\Benchmark 52
Stored procedure configuration 56
RPC / DTAQ Configuration tool 57
Program description 58
DataQueues description 59
Common Delphi problems 64
Using Language Conversion Tables 64

Known Limitations 65
16 bits 65
Technical Support Advanced Client Easycom/400 for Delphi 68
In North America 68
In Europe and other countries 68

CHAPTER 1

INTRODUCTION

Introduction

Advanced Client Easycom/400 for Delphi is a complete and powerful AS/400 development tool. It is a high-level performance middleware to access all AS/400 resources and services in a native mode.

- **Advanced Client Easycom/400 for Delphi** is an integrated database access in the Borland Database Engine, making the access to AS/400 resources fully integrated with Borland's Delphi.
- The IDAPI driver part of **Advanced Client Easycom/400 for Delphi** opens the access to AS/400 data through the standard Delphi components (TTable, TQuery, TDatabase) by using alias definition.
- **Advanced Client Easycom/400 for Delphi** allows other tools included with Delphi, such as Report Smith, to gain transparent access to the AS/400 data.
- **Advanced Client Easycom/400 for Delphi** includes a full record level access over any protocol including TCP/IP.
- **Advanced Client Easycom/400 for Delphi** shields the developers from the complexity of designing and calling API's directly.
- Accessing data is performed in native mode for all file IO operations.
- Record Blocking is optimized and completely handled in a transparent form.
- Access all AS/400 resources in batch mode including executing any command or program on the AS/400.
- Communicating with all preexistent COBOL, RPG, CL, and C, in a very simple manner.
- Access to stored procedures, data queues, and data areas.
- The security and data integrity of the AS/400 is not compromised in any way, offering Delphi developers a secure and reliable high capacity data server.
- **Advanced Client Easycom/400 for Delphi** uses the Easycom engine that was proved over the years as one of the best integration of visual component access to AS/400 data.

Performance

Using **Advanced Client Easycom/400 for Delphi**, Delphi applications can have complete real time access to AS/400 resources, without any data replication.

Advanced Client Easycom/400 for Delphi database-related features include

Files can be read from, written to and updated whether they are physical, logical or 36 format. In order to provide optimum performance, **Advanced Client Easycom/400 for Delphi** uses existing access paths and uses native mode access only.

- Delphi programs use logical files as indexes
- No code is generated on the AS/400
- Database integrity constraints are fully respected
- Access to query files
- Setup and query of databases (OPNQRYF, OVRDBF, etc.)
- Simultaneous access to multiple AS/400s
- File sharing and record level locking
- Access to Commitment Control
- Record Level access to data

ACE/400 for Delphi resource-related features include

All AS/400 resources can be accessed from within Delphi. Programs can be called from within Delphi, parameters can be passed, and the results can retrieve back in the PC environment.

- Submit a command and execute on AS/400.
- Invoke AS/400 programs and retrieve returned parameters.
- Access to data areas and data queues.
- Access to AS/400 jobs
- Access to communication APIs

AS/400 Communication Protocol

Advanced Client Easycom/400 for Delphi efficiency has been optimized to use a wide range of communication protocols. The following are just a few of the advanced communications features implemented:

APPC

Advanced Client Easycom/400 for Delphi ensures high performance data access to the AS/400 by using the APPC protocol. No additional communications layers are required.

Microsoft SNA/Server

ACE/400 Idapi Driver ensures high performance data access to the AS/400 by using the Microsoft SNA/Server protocol. No additional communications layers are required.

TCP/IP

TCP/IP is rapidly becoming the new standard in communication for AS/400. It provides better performances and is integrated in Windows 95 and Windows NT, so no additional product is required.

Native Mode

All database access is done in native mode for optimum access speed and to minimize resource requirements. Native mode access yields performance, which cannot be attained using DDM (Distributed Database Management), or resource intensive SQL.

Batch Reads

By using batch reads, entire sets of data can be retrieved in a single pass.

Integration

Once the AS/400 side of **Advanced Client Easycom/400 for Delphi** is installed on the AS/400, nothing else needs to be done on the AS/400.
No programs need to be compiled on the AS/400, no host programs need to be written, and resource usage is minimal.

Advanced Client Easycom/400 for Delphi uses the database description from the AS/400, in real time.

There is no duplicate copy kept on the client station. Data type conversions are done automatically, and access can be in either navigational mode or through SQL.

Delphi developers access data through the standard VCL (Visual Component Library) components, which are connected to a BDE alias.

Applications can simultaneously use data on AS/400 system(s), local files and on other servers.
For example, you could easily create an application using personnel data from the AS/400, sales orders from a Microsoft SQL Server, and commission rates from an Oracle database, xBase DBF or Paradox table.

How does it work?

Advanced Client Easycom/400 for Delphi is a Client/Server product. The Client running on the PC sends requests and receives results from a Server running on the AS/400.

Server Side

The EASYCOM database engine resides on the server. This proven technology provides a robust and reliable way to reach the AS/400 data.

The EASYCOM engine is capable of supporting multiple simultaneous clients, from multiple simultaneous client surfaces (i.e.: Delphi and Excel).

For further information, please see “EASYCOM Integration and Interoperability”.

Client Side

In 16 bits, the IDAPI01.DLL (which is part of Delphi) is renamed when the **Advanced Client Easycom/400 for Delphi** is installed.
The EASYCOM Idapi Driver DLL is called IDAPI01.DLL.

In 32 bits a native BDE driver is supplied. When the BDE makes a request for data, it goes through this DLL or driver.

If it is a request for data from an AS/400, then that request is dispatched to the EASYCOM server running on the AS/400. Data being returned is passed back in the format expected from the BDE.

AS/400 Extensions

In addition to transparent database access through the BDE, **Advanced Client Easycom/400 for Delphi** also provides Delphi developers with the ability to:

- Issue RPC (Remote Procedure Calls)
- Ability to invoke RPG or COBOL routines on the AS/400, and receive a result set
- Full access to the communications API
- Complete control of AS/400 jobs (rights, priorities, etc.)
- Communication by data queues
- Access to native AS/400 programs

EASYCOM Integration and Interoperability

Advanced Client Easycom/400 for Delphi is fully integrated with the EASYCOM communications engine, making all of the advanced AS/400 functionality available to Delphi developers.

This integration allows users to access data using Delphi applications, with simultaneous access by Excel or Word users who are using the same data.

Record-Oriented

Because **Advanced Client Easycom/400 for Delphi** is record-oriented; it lends itself well to applications, which require fast performance in a client/server environment.

Co-operative Systems

Because RPC is fully supported, you can achieve tremendous performance gains and reduced network traffic by having your AS/400 developers working with the Delphi developers.

Batch-oriented processing routines could be written to run on the AS/400 and be invoked by the Delphi applications. This optimum approach leverages the benefits of both environments, and maximizes the investment in hardware.

Incremental Phase-in

Because RPC allows creation of co-operative systems, you can start developing in Delphi and call existing routines written on the

AS/400, which makes it possible to upgrade systems on a module by module basis.

CHAPTER 2

SYSTEM REQUIREMENTS

System Requirements

AS/400 System Requirements

- Any AS/400 series from the initial B series.
- OS/400 version 2.2 or later
- PC connections through an Ethernet, Token Ring, Twinax or SDLC network, with each PC having a single connection point

PC System Requirements

- 386/25 or better processor
- TCP/IP protocol or one of the following APPC Router:
  PCS , Client Access or NS Router
  Eicon
  Netware for SAA
  Microsoft Client SNA
  Microsoft SNA Server
- Windows 3.x, Windows NT or Windows 95
- 16- or 32-bit Delphi Developer or Client/Server Edition.

Product features

- Functions library (DLL),
- Objects library on the AS/400 (EASYCOM Technology),
- Examples of applications,
- User guide

CHAPTER 3

INSTALLATION STEPS

AS/400 Installation Steps for APPC

During the installation on the AS/400, the program will:

- Create an **EASYCOM** library
- Create files in this library
- Send and receive Network SNA files (SNDNETF, RCVNETF)

From Windows launch SRVTRF16.EXE or SRVTRF32.EXE located in the setup directory.
Shortcuts have been created in theACE400 program group.

Important notices:

The installation procedure is based on the same protocol used by **Advanced Client Easycom/400 for Delphi** to communicate between Delphi and the AS/400.

Make sure you have a router up and running before trying to install file on the AS/400 from the PC.
This installation procedure does not rely on classical File Transfer routines. It runs under Client Access, SNA, SAA, NetSoft...

Before running the AS/400 transfer, which will install the Easycom server on your AS/400, please ensure that you are logged on with QSECOFR rights.

AS/400 Installation Steps for TCP/IP

Installation using TCP/IP is performed with the FTP protocol. In this case a FTP server on the AS/400 is required (for installation process only). It is also possible to install with an APPC router and then switch to a TCP/IP connection.

If you used our FTP TCP/IP Server installation, CFGEACTCP command will be automatically launched. Otherwise, you can launch it by hand. This command will:

- Create JOBQ in EASYCOM system
- Create JOBQ in EASYCOM library
- Create the Class : Easycom/EACCLS (*CLS)
- Create the sub system : Easycom/EASYCOM (*SBSD)

- Copy the object : STREACD, type *PGM and *CMD to the library QGPL.

- Start the sub system Easycom with the job EASYCOMD on it.

The object STREACD must be copied in a library of the *LIBL.
By the command CFGEACTCP, you can replace QGPL with any other name (Strike F4).

You can modify the objects generated by CFGEACTCP to tune your system.

The Easycom Jobs will start according to the description Easycom/EACJOBD.
If Easycom/EACJOBD does not exist, the jobs start according to the job description of the user, or according to QDFTJOBD.
By default it uses the priority class EACCLS defined in the installation library.

Notice: EASYCOM is the default library name. It could be changed during installation process.

After each IPL, you have to

- Start the Easycom/EASYCOM sub system :
  STRSBS Easycom/EASYCOM

This start can be included in the autostart of your system for next IPL.

By default, the jobs started using TCP/IP will run in QBATCH subsystem.

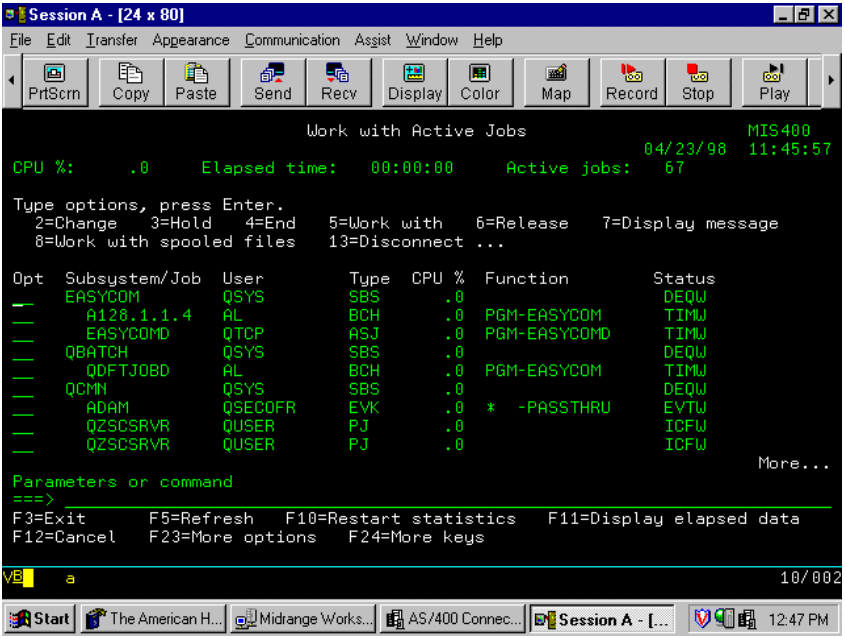
You can configure your system to run the jobs in another subsystem. In that case, you have to create a JOBD named ACJOBD in the Easycom library. EACJOBD$ is given as a sample of the JOBD.

In case of troubles:

Check if the EASYCOMD job is running in the Easycom subsystem. This job is started by STREACD.

If you are running in the QBATCH subsystem, check the maximum activity of the Subsystem. May be your job is waiting in the JOBQ.

Once the JOBQ EACJOBQ exists in the EASYCOM Library, this JOBQ must be added to an active subsystem, and the routing data and class have to be configured correctly.


Session A - [24 x 80]
File Edit Transfer Appearance Communication Assist Window Help
PrtScrn Copy Paste Send Recv Display Color Map Record Stop Play
Work with Active Jobs
MIS400
04/23/98 11:45:57
CPU %: .0 Elapsed time: 00:00:00 Active jobs: 67
Type options, press Enter.
2=Change 3=Hold 4=End 5=Work with 6=Release 7=Display message
8=Work with spooled files 13=Disconnect ...
Opt Subsystem/Job User Type CPU % Function Status
EASYCOM QSYS SBS .0 DEQW
A128.1.1.4 AL BCH .0 PGM-EASYCOM TIMW
EASYCOMD QTCP ASJ .0 PGM-EASYCOMD TIMW
QBATCH QSYS SBS .0 DEQW
QDFTJOBD AL BCH .0 PGM-EASYCOM TIMW
QCMN QSYS SBS .0 DEQW
ADAM QSECOFR EVK .0 * -PASSTHRU EVTW
QZSCSRVR QUSER PJ .0 ICFW
QZSCSRVR QUSER PJ .0 ICFW
More...
Parameters or command
===>
F3=Exit F5=Refresh F10=Restart statistics F11=Display elapsed data
F12=Cancel F23=More options F24=More keys
a
10/002
Start
The American H...
Midrange Works...
AS/400 Connec...
Session A - [...
12:47 PM

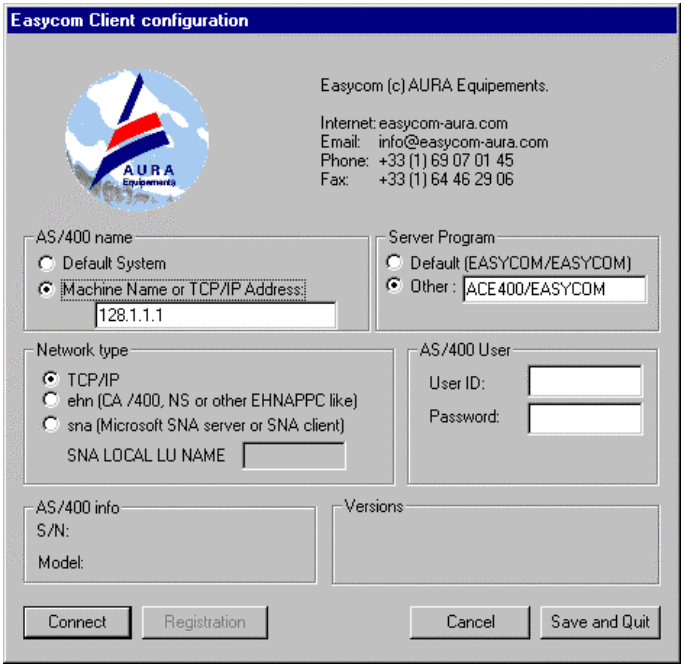
ACE400 will automatically use the PC station name to identify the job on the subsystem. When the station id is not found then the IP address will be used.

To monitor the jobs executed on the AS/400, operations could be reviewing the jobs in the Easycom subsystem.

AS/400 Activating the Serialization key

From the EASYCOM program group, select and launch the EASYCOM Configuration (16- or 32-bit).

Fill the following boxes:



AS/400 Name:

- **Default System**
- **Machine Name or TCP/IP address:**

Select the default system if you are using a single AS/400, or select and fill in the Machine Name.

If the router is not connected by default to the AS/400 you want to use with **Advanced Client Easycom/400 for Delphi**, you need to set the correct name. If you are using SNA Server, the AS/400 is known by its alias name, defined in the SNA Server configuration.

Server Program:

- **Default (EASYCOM/EASYCOM)**
- **Other:**

Select the default system or type in the library name you want to use. You can use a library name other than EASYCOM.

Network Type:

- **TCP/IP**
- **SNA (Microsoft SNA Server or SNA Client)**
- **EHN (CA/400, NetSoft or other EHNAPPC type)**

Select TCP/IP if you are using TCP/IP, SNA if you are using SNA or SAA and EHN in all other case.

NoWait Mode:

- **Enabled (Default)**
- **Disabled**

Select the default option. Disabled mode can be useful in very particular situations.

Local LU Alias:

This field is valid only with SNA Server. Your SNA Server administrator has defined several LU Aliases (Local Unit). If you don't have a default LU assigned for your user name, or your group, you have to give an existing LU name.

This information will be stored in the EASYCOM.INI file.

Click on the **connect** button.

If the connection is successful, the **connect** button will be disabled and the **registration** button enabled.

If you receive an error dialog box like "EASYCOM.DLL Router not installed", you are experiencing a connection error and you must retest all aspects of your connection.

Notice: You can check what version of **Advanced Client Easycom/400 for Delphi** you are using. In the main windows of the Easycom serialization you have a Version section.

Into **Advanced Client Easycom/400 for Delphi** build 3.0,you get:

For 16 Bits systems:
- CLT: 4.02.02
- SRV: 4.52.00
- DRV: 3.0.10

For 32 Bits systems:
- CLT: 4.02.02
- SRV: 4.52.00
- DRV: NA

Delphi Installation Steps

There are no specific installation steps required for **Advanced Client Easycom/400 for Delphi**. A new alias named “ACE400” of type ACE400 is created.
Set the “library name” to the library name in the AS/400.

If you want to use ReportSmith in 16 bits, change the type of the alias from “ACE400” to “INTRBASE”.

Alias configuration options

These are main configuration options.

SERVER NAME

This is the name (or IP address) of your AS/400. It is possible to change it at run time, and connect to multiple AS/400 at the same time.

LIBRARY NAME

This is the default working library. If in your applications you don't name the library,the value will be the default one.
Valid values are:

- any existing library name on your AS/400
- *LIBL
- *USRLIBL (default)
- *CURLIB

BROWSE LIBS

During design time, a special dialog box prompts allowing you to browse the files of your AS/400.

If you don't want to see this dialog box, change this option to FALSE. The files will be the files of the library specified by 'LIBRARY NAME' option.
Default value is TRUE.

LANGDRIVER

This is the default language driver for your PC. Default is ANSI.

CODEPAGE FILE

This is the name of a codepage file, to allow custom EBCDIC-ASCII conversions. You can choose any file from the files in the 'codepage' subdirectory of the installation.

Be careful: you must specify the FULL PATH of the file.
Default is e297ansi.cpg.

To know which is the codepage file for your AS/400, make a DSPSYSVAL QCCSID on a green terminal.

Suppose that this is 285, and you have installed the product on C:\ACE400. You will have to put this for 'CODEPAGE FILE' option:
C:\ACE400\CODEPAGE\e285ansi.cpg
'285' is the codepage of your AS/400 'ANSI' is the codepage for your PC.

CMTCTL

Set it to TRUE if you want to work with transactions.
You will be able to use BeginTransaction,
Default is FALSE.

PRIVATE SESSION

Set it to TRUE if you want that each connection (by a Tdatabase Component) must be one different connection on the same AS/400.
Default is FALSE.

LOG LEVEL

This allows you to make a text trace of operations that are made by the server. If 0, no trace is made.
Default is 0.

LOG FILE

Represents the file on the AS/400 that will be created if LOG LEVEL is not zero.

Misc. configuration options

Additional configuration options are provided, for special behaviors. Some of them are only provided for backward compatibility with LightLib/400.

These options are in easycom.ini file.

- - Oldconversion, in [general] section.
  This allows to use old conversion behavior that was in LightLib/400.
  The numeric fields (packed or zoned) will always be converted into double values or character values (if above the precision of double).
  Put Oldconversion=1 to enable this.
  Default is 0

- - ThreadAttach in [Multithread] section and NbSession in [NT] section.

By default, with multithread, the AS/400 connections are shared with the threads.

To automatically generate one connection per thread, add in easycom.ini file:
[MultiThread]
ThreadAttach=1
[NT]
NbSession=10

Default is 0 for ThreadAttach and 1 for NbSession.
Nbsession is the maximum number of AS/400 connections. It can be up to 32.

Important note:

Be aware of TBatchMove component limitations.

TBatchMove does not create indexes for the destination table, and field types are automatically matched with source field types, which can lead to problems trying to create non-supported field types in the AS/400 or transferring empty date fields.

You will probably use native AS/400 tools to create and manage AS/400 tables, but we wanted to provide you with maximum compatibility with existing Delphi components.

CHAPTER 4

CONNECTIONS

Testing Data Access

Before running the **Advanced Client Easycom/400 for Delphi** samples, you must be sure your PC is properly connected to the AS/400.

If you were able to run the EASYCOM installation and transfer files onto the AS/400, this can be taken as a good sign, but it's not a guarantee.
You could still experiment connection errors, depending on the setup of drivers and the AS/400 communications layer.

Launch Delphi. Create a New Application.
Drop a TTable component onto the main form.
Set the DatabaseName to ACE400.
Set the Table Name to SP_CUST in the EASYCOM library using the Dialog. Once you select it, a "real" connection will be established to the AS/400, which could take a while. When prompted for a Database password, press enter.

Note: The dialog fills the combo box list of the Table Name property. Once this is done, you still need to select one of the items.

Set the Active property to True.

Drop a TDataSource component and connect it to the TTable.
Drop a Grid component, connect it to the DataSource, you should see the AS/400 data in the grid.

If you have been successful with these steps, then congratulations, you have just created your first Delphi AS/400 application! You are ready to move on and do some real work with the AS/400 and Delphi. But first, we would suggest that you first take a look at the sample applications provided with **Advanced Client Easycom/400 for Delphi**.

Please notice that the Logical files associated to a physical file are visible in the IndexName and IndexFields.

If you did not get data displayed in a grid, then there's no point in trying the samples as they will fail.
Carefully check all of the installation steps, on the Delphi side (aliases ...), then, on the AS/400 side, check if you are receiving any error messages, and check the connectivity layers.

Use the Benchmark and Serialization programs to retest communications.

Field type management

All kinds of AS/400 field types are supported. For the field types that don't exist on AS/400, they are converted to the nearest field type.

Null capability is also fully supported: you can get or set null field information with tables and queries. You can even make a key search with null values.

Depending of the AS/400 type of a field, it is converted (in read and write) to a PC type. Conversion between AS/400 types and PC types are made by the client.

The conversion map follows:

AS/400 type	**Size**	**Dec**	**PC Type**
CHAR	Any		ZString (pChar)
BIN2			INT16
BIN4			INT32
PACK or ZONED	1->4	0	INT16
PACK or ZONED	5->9	0	INT32
PACK or ZONED	10->15	Any	DOUBLE
PACK or ZONED	Other cases		ZString (pChar)
DOUBLE			DOUBLE
FLOAT			DOUBLE
DATE			DATE
TIME			TIME
TIMESTAMP			DATETIME

You can notice that PACKED or ZONED fields are optimally mapped to a type depending on the size.

CHAPTER 5

EXAMPLES

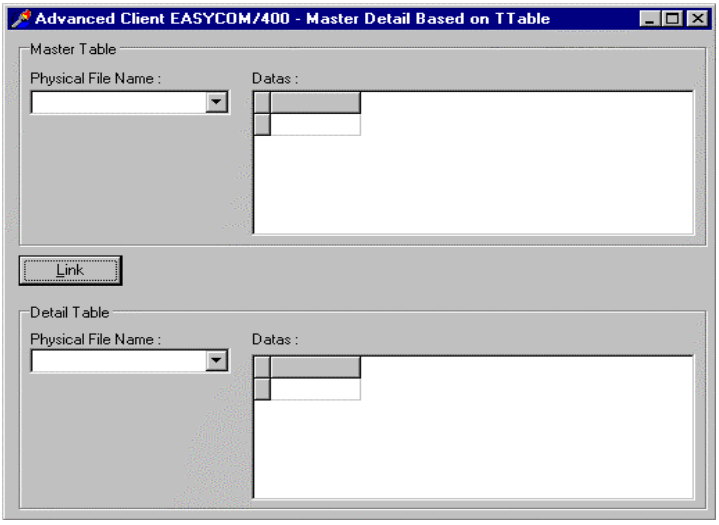
Running the Sample Programs

Some sample Delphi programs have been installed in your ACE400\SAMPLES sub-directory. These samples demonstrate some of **Advanced Client Easycom/400 for Delphi**'s features and capabilities.

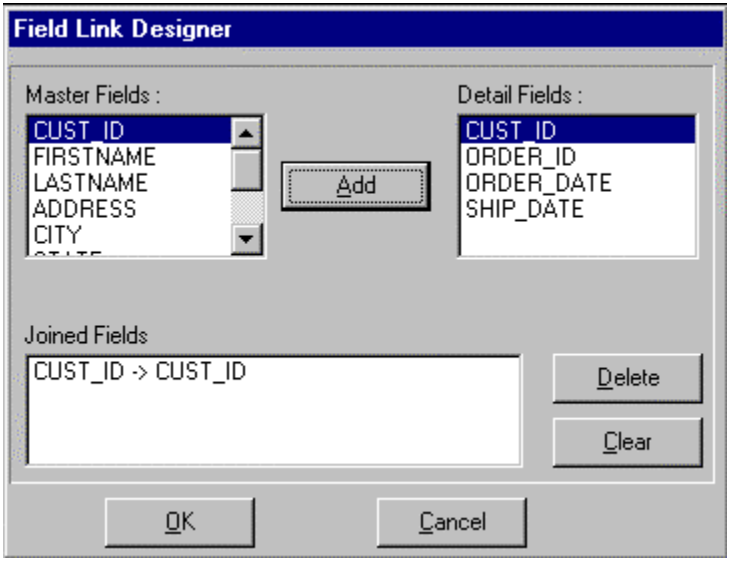
Please note that some samples are 32-bits only.

Delphi\MasDetTb\ (16 and 32 bit):

In this sample we are given a chance to look at two tables in a database. When you start the program it will look like below:



The top look-up is for choosing the master table. The bottom is for choosing the detail table. The grids will show the respective tables. For now we shall choose SP_CUST for the master table, and SP_ORD for the detail table. Once you have choosen the tables click on Link. The following form should appear:



The lists show the fields in the two tables. Left being the master table and the right being the detail table. Click on the field that is common between the two, CUST_ID. Click on Add. The grid on the bottom shows the fields relationship. Click on OK.

What you now shall see is that the detail grid will only show those fields that pertain to the customer that the master field is pointing to.

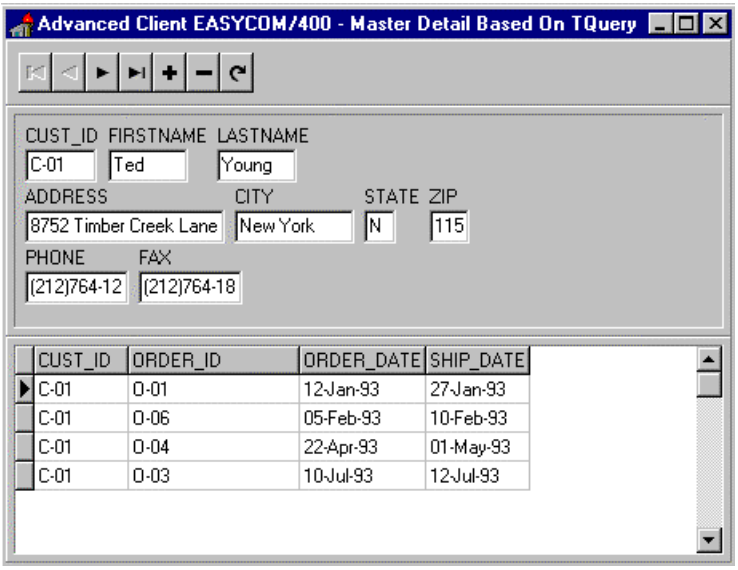
Delphi\MasDetTQ\ (16- and 32-bit):

The second sample shows you the compatibility between **Advanced Client Easycom/400 for Delphi** and the TQuery component.

The query syntax submited has to comply with AS/400 Query syntax requirements. ACE/400 submits the supplied Query to the AS/400 query engine to be performed. So, If you intend to make extensive use of TQuery, you should consider using **Advanced Client Easycom/400 for Delphi**.

The second sample shows you the compatibility between EASYCOM Idapi Driver and the TQuery component.
Even if this is not the fastest way to access the AS/400 (because in this case the SQL engine of the AS/400 which wraps the Physical/Logical files layout will be activated on the AS/400), it is good to know you have a complete compatibility with the TQuery component.

Running this example as is will give you DBEdits and a grid show the prevelant information.



Similar to the MasterSettb sample, we see the master table, customer information, and the detail table, the order information, and their relationship through CUST_ID.

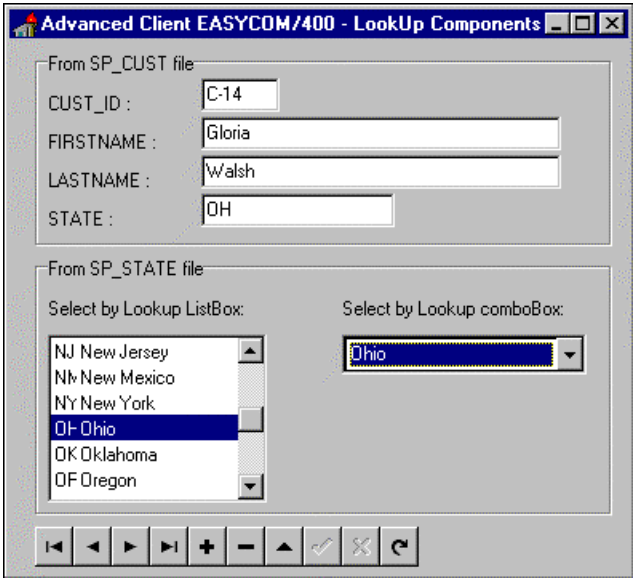
This sample does allow for editing, adding, and deleting of records. Try it.

The MastDetTQ sample was created using the Delphi Master/Detail Form Expert, using a TQuery. It uses SP_CUST and SP_ORD, the CUST_ID field as MasterKey, but no Logical Files on the AS/400, which accounts for the slower performance.

Delphi\LookUp\ (16- and 32-bit):

This sample shows you the compatibility between **Advanced Client Easycom/400 for Delphi** and DBLookUpListBox and DBLookUpComboBox.
It is based on two files with logical files.

Again, we are showing a master-detain relationship. The master table is SP_CUST giving all the customer information. The detail table happens to be SP_STATE, which gives the state name to state code.

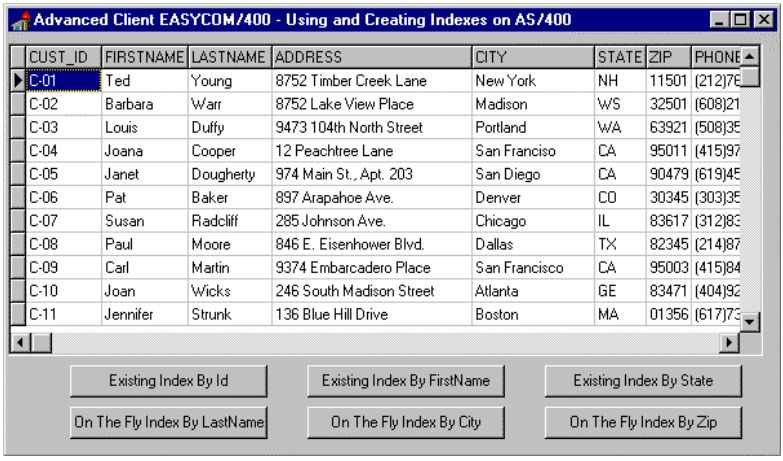


You travel through the records using the navigator. What the lookupbox and lookupcombo allows you to do is change the state in the SP_CUST table. Inserting and deleting records is also poosble in this sample.

Delphi\FlyIndex\ (16- and 32-bit):

The FlyIndex sample demonstrates how to use logical files as indexes, as well as a direct call to the AS/400 database system through the API in order to create temporary indexes on the fly.

This sample uses SP_CUST to show the table in acsending order by whatever choice the user wants. The user might choose an already existing field or describe is own.

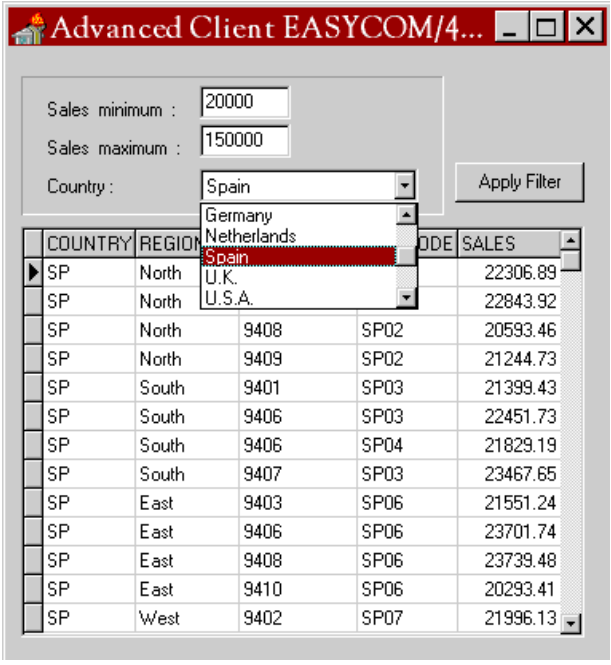

Advanced Client EASYCOM/400 - Using and Creating Indexes on AS/400
CUST_ID FIRSTNAME LASTNAME ADDRESS CITY STATE ZIP PHONE
C-01 Ted Young 8752 Timber Creek Lane New York NH 11501 (212)7
C-02 Barbara Warr 8752 Lake View Place Madison WS 32501 (608)21
C-03 Louis Duffy 9473 104th North Street Portland WA 63921 (508)35
C-04 Joana Cooper 12 Peachtree Lane San Franciso CA 95011 (415)97
C-05 Janet Dougherty 974 Main St., Apt. 203 San Diego CA 90479 (619)45
C-06 Pat Baker 897 Arapahoe Ave. Denver CO 30345 (303)35
C-07 Susan Radcliff 285 Johnson Ave. Chicago IL 83617 (312)83
C-08 Paul Moore 846 E. Eisenhower Blvd. Dallas TX 82345 (214)87
C-09 Carl Martin 9374 Embarcadero Place San Francisco CA 95003 (415)84
C-10 Joan Wicks 246 South Madison Street Atlanta GE 83471 (404)92
C-11 Jennifer Strunk 136 Blue Hill Drive Boston MA 01356 (617)73
Existing Index By Id
Existing Index By FirstName
Existing Index By State
On The Fly Index By LastName
On The Fly Index By City
On The Fly Index By Zip

Delphi32\Filters (32-bit):

The Filters sample demonstrates how to do on the fly filtering of a TTable.

It uses standard Delphi properties and methods and allows you to change the filter applied to a table on the fly.

Filtering in this sample is based on both a range of values for sales and a country code.



This particlar sample only takes to tables and a button to filter the files. What helps is the following code :

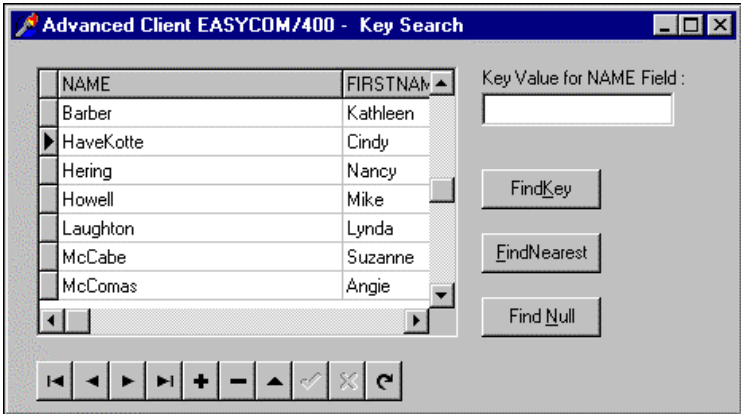
Table1.Filtered := False;
Table1.Filter := 'SALES>=' + SalesMin.Text + ' AND SALES<=' + SalesMax.Text;
Table1.Filter := Table1.Filter + ' AND COUNTRY=' + "" + DBLookUpComboBox1.KeyValue + "";
Table1.Filtered := True ;

This tells the program to filter Table1 and how to filter it.

Delphi\Find (16 and 32 bit):

The Find Sample uses the Find methods on a key field. You can find a value in the FIRSTNAME field or a nearest value. You can also search a null Value.

This is from a Table in Easycom called People. The buttons are user-defined that call on pre-existing procedures. Find Key uses the FindKey procedure, which looks for an exact match. Find Nearest calls on FindNearest, which looks for the closest match. Find Null calls on FindKey passing the NULL character as a parameter.



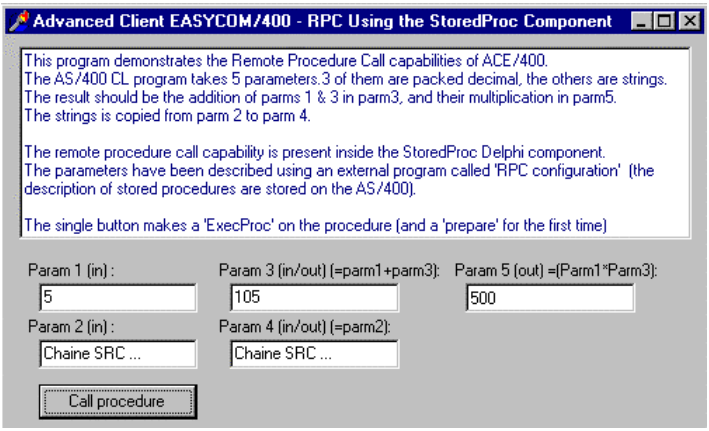
Delphi\StProc (16 and 32 bit):

This sample demonstrates the Remote Procedure Call capabilities of **Advanced Client Easycom/400 For Delphi**.

The AS/400 CL program takes 5 parameters.

3 of them are packed decimal, the others are strings.
The result should be the addition of parms 1 & 3 in parm3, and their multiplication in parm5.

The strings is copied from parm 2 to parm 4.


Advanced Client EASYCOM/400 - RPC Using the StoredProc Component
This program demonstrates the Remote Procedure Call capabilities of ACE/400.
The AS/400 CL program takes 5 parameters.3 of them are packed decimal, the others are strings.
The result should be the addition of parms 1 & 3 in parm3, and their multiplication in parm5.
The strings is copied from parm 2 to parm 4.
The remote procedure call capability is present inside the StoredProc Delphi component.
The parameters have been described using an external program called 'RPC configuration' (the description of stored procedures are stored on the AS/400).
The single button makes a 'ExecProc' on the procedure (and a 'prepare' for the first time)
Param 1 (in) :
5
Param 3 (in/out) (=parm1+parm3):
105
Param 5 (out) =(Parm1*Parm3):
500
Param 2 (in) :
Chaine SRC ...
Param 4 (in/out) (=parm2):
Chaine SRC ...
Call procedure

The remote procedure call capability is present inside the StoredProc Delphi component.

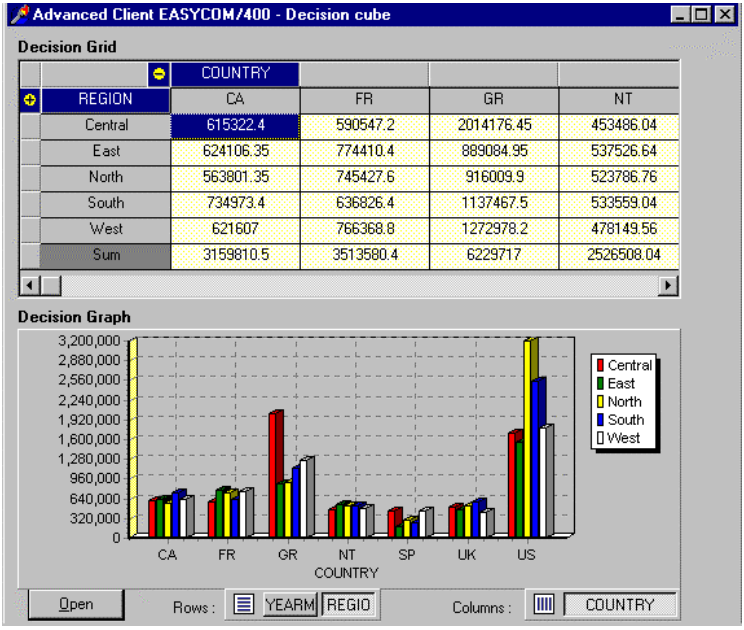
The parameters have been described using an external program called “RPC configuration” (the description of stored procedures are stored on the AS/400).

The single button makes a “ExecProc” on the procedure (and a “prepare” for the first time)

Delphi3\DCube (32 bit, Delphi 3 only):

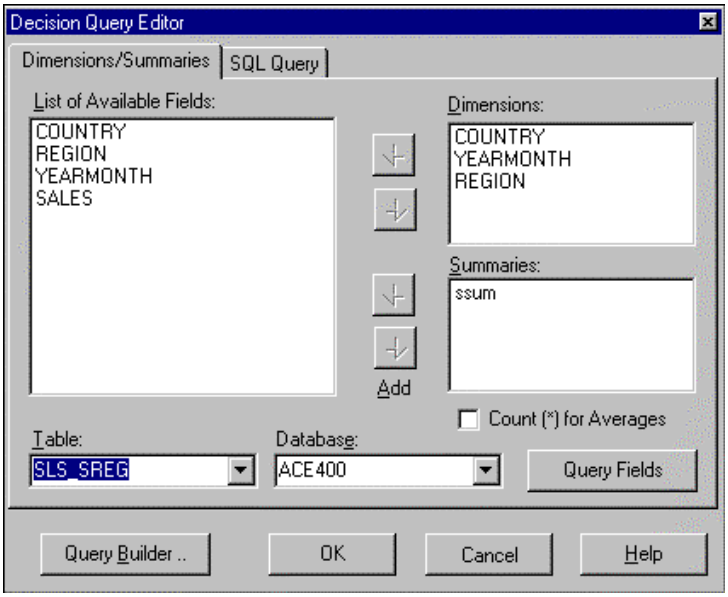
This sample is based on Decision Query. You can see a Decision Grid and a Decision Graph.

Running the sample will give you the following screen:



Click on Open to start the sample. The decision grid show the table the graph is based off of. The buttons on the botom: YEARM, REGION, and COUNTRY, allow you to choose which fields you want to influence the graph.

To change the database you want to view, go into Delphi, and open the sample. Open the form and right click on the Decision Query. Select the Decision Query Editor. This will bring up the following form:

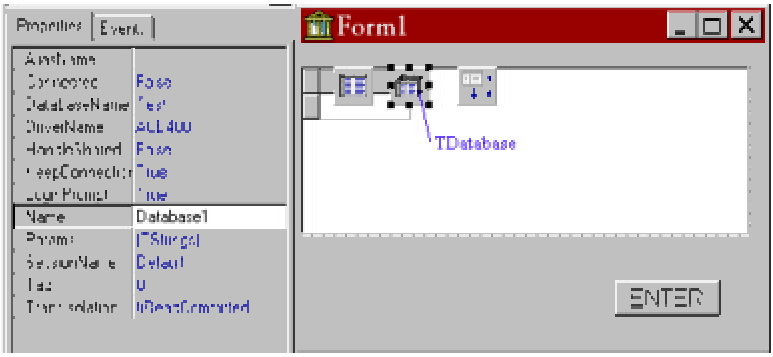


From here, you can choose the Database, Table, which fields to use, the sums of fields, or customize it personally. This query builds the following lines automatically:

```
SELECT COUNTRY, YEARMONTH, REGION, SUM( SALES ) ssum
FROM SLS_SREG
GROUP BY COUNTRY, YEARMONTH, REGION
```

Delphi3\NoLogIn

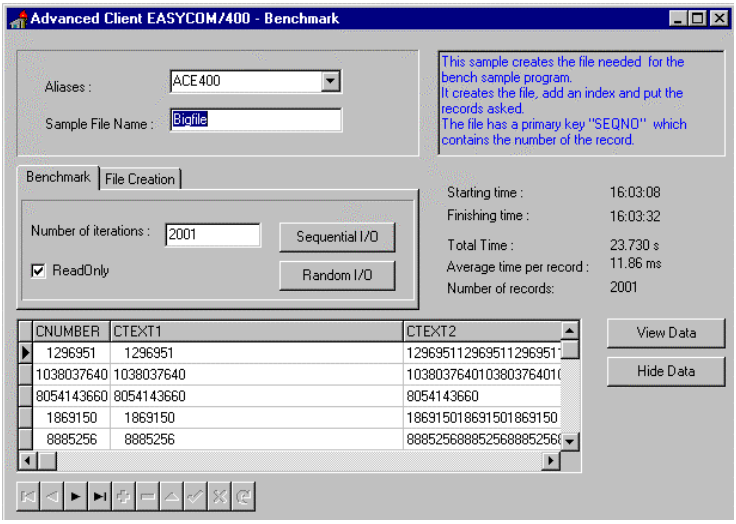
This sample automatically logs you into a databse your trying to get information from. This keeps you from having to log in everytime you want to do something.



The way to do this sample is to take a Tdatabase, pass the login process as parameters. This sets up a temporary database in the program. Now all you have to do is set your table(s) to the temporary database.

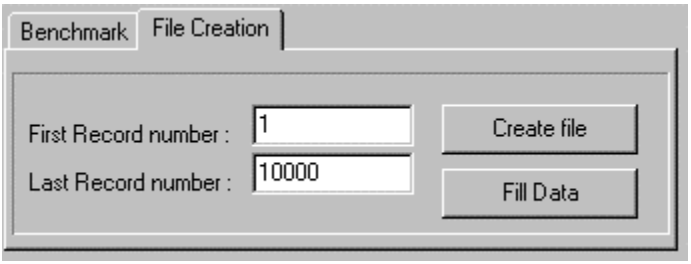
Delphi3\Benchmark

This sample's main purpose is to check the size of a table, the number of records, and how long it takes to transverse those records. It was also designed to create files to run tests on.



The aliases on top refers to database names. Sample file name determines which table the program will look into. In this sample, we went with Bigfile.

The benchmark has two buttons, Sequential and Random I/O. Sequentil refers to starting from the beginning and going in order to the end. Random refers to randomly picking the records. Number of Iterations determines how many of the records will be read. The ReadOnly check refers to whether you want to add, delete, or edit the records.



Under the File Creation, there are the Create file and Fill Data buttons. They create a test file and the fill it full of data. The first Record number is the index number the table will start at. The last Record number is the last index number.

There is also a record of how long it takes to transverse the records. This tells you the starting time, finish time, total time to transverse the records, average time per record, and the total number of records. Also are two buttons that either turns the grid on or off.

To run it, choose your database. Either choose a table to test, or make your own. You can transverse the records using squential or random I/O. Try both, see if there is a time difference between the two. Don't forget, you can edit the records as you go along.

CHAPTER 6

AS/400 FUNCTIONS

Stored procedure configuration.

AS/400 programs do not have any external description.
The program expects data in parameters, and returns results through the same parameters.

Only the program itself knows about the type and size of those parameters.

You need to describe your program, **because Advanced Client Easycom/400 for Delphi** needs to know how to translate the information from AS/400 into PC format.

If your program has a variable structure depending on the value of a parameter during the CALL, you can provide multiple descriptions for the same program.
Each description will have its own procedure name.

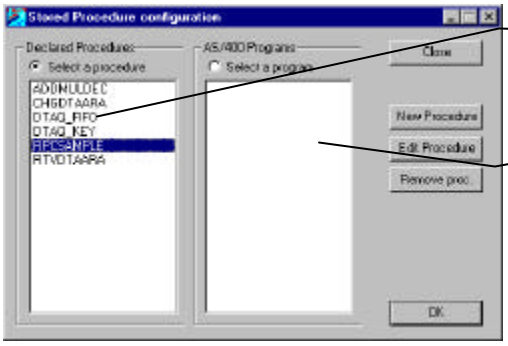
The EASYCOM Stored Procedure Configuration tool allows you to describe any program on the AS/400 by giving the type and size of each parameter it expects to receive.

You can describe any AS/400 program written in any AS/400 language (RPG, COBOL, CL, C, etc.).

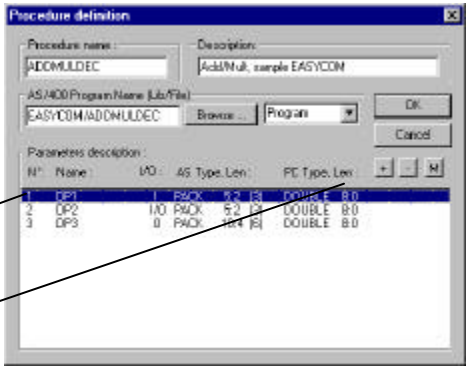
The only restriction is that the program must not do any screen or keyboard I/O, because it will be treated as a stored procedure, and will be an extension of the Easycom server.

RPC / DTAQ Configuration tool

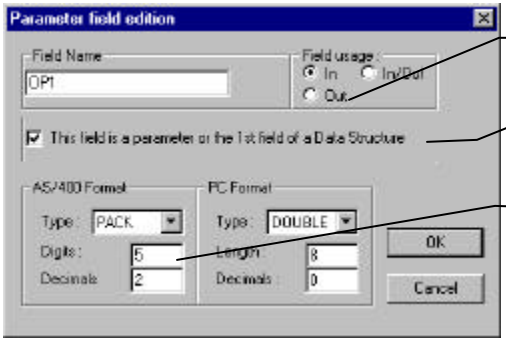
You can start the RPC/DTAQ Configuration tool from the shortcut in the program group “ACE400”



- *Existing procedures*
- *Existing program or DataQueue*



- *Program or DataQueue localization on the AS/400*
- *Type (Program or*



- *Field Use*
- *Field type*
- *Field format both on the AS/400 and on the PC*

Program description.

❶ Start the « EASYCOM Stored Proc Configuration » , in the program group « ACE400 »,

❷ Create a new procedure

❸ Complete the dialog box with the Procedure description:

- **Procedure Name**: This name will be the only one used on the client side. Your application will open and call this procedure name.
- **Description**: Enter any text to describe your procedure.
- **AS/400 Program Name:** The complete AS/400 qualified program name. LIBRARY/PROGRAM
- **Program/DataQueue:** Leave this Combo Box in Program state.

❹ Describe the parameters expected by your program.

Because on AS/400, a parameter can be a Data Structure, you need to describe each Field (item of the Data Structure), and to put the type of the field (parameter or first field of a Data Structure or not).

You can Add, Delete or Modify field descriptions by clicking the "+", "-" or "M" buttons.

Complete the next Dialog box " Parameter field edition ":

- **Field Name:** The field to be used
- **Field usage:** The field is sent to the program (IN), or it is the return result of the program (OUT) or both. Usually the parameters are IN/OUT.
- **This field is a parameter or the first field of a Data Structure:** This box must be unchecked only if the field is in the second or greater position inside a parameter that is a Data Structure.
- **AS/400 format:** Describe exactly which data format your program expects: Data Type, Number of characters or digits for a decimal value, and the number of decimals.
- **PC format:** Describe how you want the data to be converted by Easycom.

You can use this process to describe several programs, also you can provide multiple descriptions for the same program, if needed. See StProc sample provided for a complete example.

DataQueues description

You can also create procedures for all type of known AS/400 Data Queues : simple file (FIFO) or keyed file (KEYED).

“DTAQ_FIFO” is an example based on simple dataqueue while “DTAQ_KEY” is an example with keyed procedure

❶ Start the “ Easycom Stored Proc Configuration ” in the program group “ ACE400 ”,

❷ In the “ Stored Procedure configuration ” window, select “ New Procedure ”,

❸ Complete the dialog box “ Procedure definition ":

- **Procedure Name** : This name will be the only one used on the client side.
- **Description** : Enter any text to describe your DataQueue,
- **AS/400 Progam Name** : Path (<Library>/ <Program>) to the DataQueue,
- **Program/DataQueue** : “DataQueue”.

❹ Describe the fields of your Data Queue. Using “ **+** ”, “ **–** ” or “ **M** ” buttons, you can add, delete or modify fields

In the description of the DataQueue format, you have to create three fields used by **Advanced Client Easycom/400 for Delphi**.

- TIMEOUT : ‘char’ type , size 6 Time-out reading delay,
- FILER : ‘char’ type , size 2 Future use,
- ORDER : ‘char’ type, size 4 Kind of key operation (DataQueue KEYED),

In case of Keyed Data Queue, fourth field is the key

Complete the next Dialog box " Parameter field edition ":

- **Field Name** : Virtual name of the field,
- **Field usage** : No meaning for the DataQueue.
- **This field is a parameter or the 1st of a Data Structure** : No meaning for the DataQueue,
- **AS/400 format** : Describe exactly which data format your program expects: Data Type, Number of characters or digits for a decimal value, and the number of decimals,
- **PC format** : Describe how you want the data to be converted by Easycom.

You can use this process to describe several Data Queues, and you can provide multiple descriptions for the same Data Queue, if needed.

CHAPTER 7

COMMON PROBLEMS

Common Delphi problems

Delphi 16 users: Please be aware of the 256 character path limitation.
The Delphi16/VCL directory must be in your path, if you exceed the path limit you'll need to map a drive or consolidate VCLs in a separate directory to ensure they are all found.

Using Language Conversion Tables

In some countries (i.e. Israel), you would need to use translation table. This is the case when the AS/400 or the PC does not produce US/LATIN EBCDIC or ANSI characters. This will allow Delphi DataAware controls to work properly.

To use the conversion table, modify the “CODEPAGE FILE” in the alias configuration as follow:
CODEPAGEFILE …MyDirectory\MyPath\MyConversionTableName

Advanced Client Easycom/400 for Delphi comes with a set of conversion tables. They all follow the naming convention:

Table name convention:
EnnnAmmm.tbl
Where
nnn is the EBCDIC code page.
mmm is Ascii code page. Ammm can be replaced by ANSI.

The contain of a table is :

Bytes 0 to 255	= EBCDIC to ASCII Conversion.
Bytes 256 to 511	= ASCII to EBCDIC Conversion.

Ex : C:\ACE400\CodePage\e297ansi.cpg

The value of the character will be used as an index in the table.

Example:
To translate character '0':
'0' = 0xF0 (240) in EBCDIC
'0' = 0x30 (48) in ASCII
Offset 240 of the file contains value 48.
Offset 560 (512+48) contains value 240.

If the alias is modified in such a way that the translation does not generate an ANSI character set, the 'LangDriver' in the alias has to be set properly, using Borland's tables (use BDE administrator).

As an example, use the table ExxxAyyy.TBL. If your PC is using a Windows code page yyy and your AS/400 is using an EBCDIC code page xxx.
In such a case you will need to set the 'LangDriver' accordingly.

Known Limitations

16 bits

Using the Database Desktop to access AS/400 table will result in a GPF.
This is due to an undocumented IDAPI function called by the Database Desktop and we don't expect this to be resolved in a future release.

Note that it does not make any sense to modify the structures of the AS/400 table using the Database Desktop instead of the AS/400 native tools as most of the AS/400 tables features are not directly mapped to PC equivalent.

TBatchMove has been successfully tested in a lot of situations (see MoveFile sample).

However, if the file already exists in the AS/400 you will get an execution error.

This is not the standard behavior of the TbatchMove component as it should override the existing table, however it is a pretty safe protection against wrong manipulations.

CHAPTER 8

TECHNICAL SUPPORT

Technical Support Advanced Client Easycom/400 for Delphi

In North America

MIS Systems Corp.
208 Third Ave. North #200
Nashville, TN 37201 USA
Phone: (615) 248-0079
Fax: (615) 248-8079
Email: support@missystems.com
Web: www.missystems.com

In Europe and other countries

AURA Equipements
Z.A DE COURTABOEUF
10, Avenue du Quebec - BP 519
F - 91946 LES ULIS CEDEX FRANCE
Phone: 33 1 69 07 01 45
Fax: 33 1 64 46 29 06
Email: tech@easycom-aura.com
Web: www.easycom-aura.com